

Serial No. 09/456,894

REMARKS

The Applicant and the undersigned thank Examiner El-Chanti for the careful review of this application. Claims 1-58 have been rejected by the Examiner. Upon entry of this amendment, Claims 1-26, 28-30, and 33-40, 42-53, and 55-57 remain pending in this application. Claims 27, 31-32, 41, 54, and 58 have been cancelled. The independent claims are Claims 1, 11, 18, 23, 26, 28, 30, 34, 38, 42, 45, 48, 51, and 55. Consideration of the present application is respectfully requested in light of the above amendments to the application and in view of the following remarks.

Objection to Claims

The Examiner objected to Claim 45 because of an informality. Specifically, the Examiner objected to Claim 45 because the phrase, "being providing", should read --being provided--.

The Applicant notes that Claim 45 was amended on March 25, 2004 in light of the Examiner's helpful comments. The Applicant believes that the Examiner may have inadvertently copied this objection from his previous Office Action since the Applicant did respond to this objection in the subsequent response. Accordingly, reconsideration and withdrawal of the objection of Claim 45 are respectfully requested.

Summary of Telephonic Interview of September 27, 2004 - Pursuant to M.P.E.P. § 713.04 and 37 C.F.R. § 1.133(b)

The Applicant appreciates the time and consideration given by Examiner El Chanti during the telephonic interview of September 27, 2004. During this telephonic interview, the claims as amended above were presented and discussed. Applicant's representative emphasized that the Examiner's primary prior art reference of U.S. Pat. No. 6,496,858 issued in the name of Frailong et al. (hereinafter, the "Frailong reference") does not provide any teaching of the code level operations that are described by the claims of the present application.

Serial No. 09/456,894

Independent Claim 1 was discussed as an example. This claim, as amended, recites (1) disabling access to at least a first section of code in a network driver interface that is being executed by the processor and (2) patching the first section of code while the network driver interface is being executed by the processor to cause the insertion of a rerouting driver into the one or more communication paths provided by the set of bindings.

Instead of disabling access to a first section of code in a network driver interface and patching the first the section of the code, the Frailong reference uses a higher, program application level, configuration manager 506 as illustrated in Figure 5 (reproduced below) which is an API operating through a remote procedure call (RPC) protocol that communicates commands between a user interface 502 and network services 512 (see Frailong reference, column 8, lines 24-40).

The Frailong reference is not concerned with reconfiguring code being executed by a processor. Further, the Frailong reference also does not provide any teaching of media access control units, protocol drivers, and bindings, as recited in amended independent Claim 1. The Frailong reference simply does not teach low level code operations below the application program level that concern code running on a processor and patching such code to include a rerouting driver while the processor is running, as recited in amended independent Claim 1. The Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program.

The Applicant's representative noted that while the nomenclature used by the Frailong reference may appear at a first glance to be relevant, after closer inspection, the higher application program level description of the Frailong technology is quite different from the low level code based operations described in the claims such as those recited in amended independent Claim 1.

At the end of the telephonic interview, Examiner El Chanti explained that he would investigate these issues further and provide the Applicant with more feedback. Applicant's representative stated that a request for continued examination (RCE) would be filed containing the amendments that were discussed at the end of the one month extension period if the Examiner would not issue a Final First Office Action. The Examiner stated that he would

Serial No. 09/456,894

likely not issue a Final First Office Action if the Applicant filed an RCE without submitting an After-Final response in view of what was discussed during this telephonic interview.

Consideration and approval of this interview summary by Examiner El Chanti are respectfully requested. The Examiner is requested to initial this summary if he approves of it pursuant to M.P.E.P. § 713.04.

Rejections under 35 U.S.C. § 102(e)

The Examiner rejected Claims 1-58 under 35 U.S.C. 102(e) as being anticipated by the Frailong reference. The Applicant respectfully offers remarks to traverse these rejections.

Independent Claim 1

The rejection of Claim 1 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest (1) disabling access to at least a first section of code that is being executed by the processor in a network driver interface, wherein the network driver interface provides for communication between one or more media access control units and one or more protocol drivers in a computer system according to a set of bindings; (2) patching the first section of code while the network driver interface is being executed by the processor to cause the insertion of a rerouting driver into the one or more communication paths provided by the set of bindings; and (3) re-enabling access to the patched first section of code, as recited in amended independent Claim 1.

The Frailong reference describes in Figure 2 (reproduced below) a gateway interface device 208 that provides a physical and logical connection between a LAN 210 and an external network, such as a remote server 206 or an internet service provider 204. The gateway interface device 208 contains a configuration manager 506 which stores the configuration information transmitted from the remote management server 206.

Serial No. 09/456,894

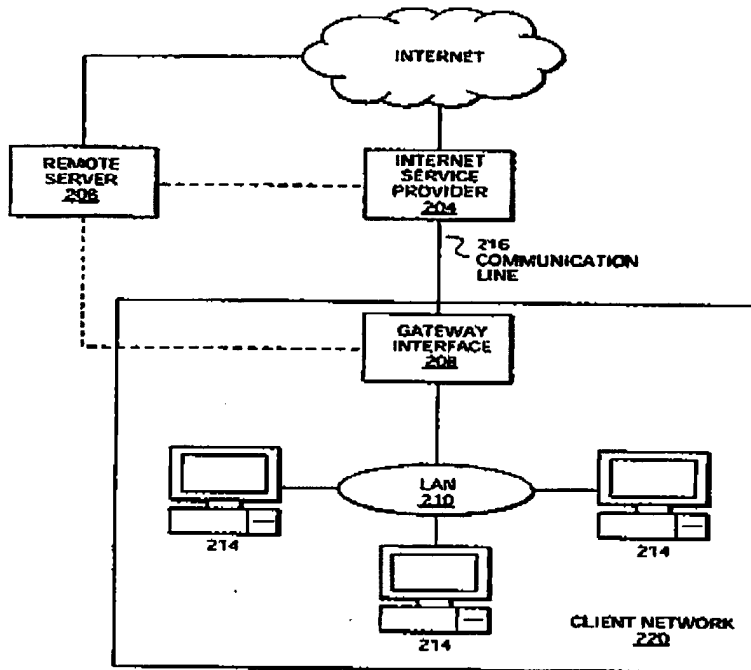


FIG. 2

Gateway interface device 208 also contains service managers 510 which communicate with network services 512 resident in the gateway interface device 208. The service managers 510 are application programming interfaces that provide the required command and data translation for various services available. See the Frailong reference, column 5, lines 40-47.

The gateway interface device 208 is a stand alone machine as evidenced by Figure 3 (reproduced below). The gateway interface device 208 includes flash memory 304, random access memory 306, read only memory 308, a mass storage device 310, a bus 302, an network interface 214, and a central processing unit.

Serial No. 09/456,894

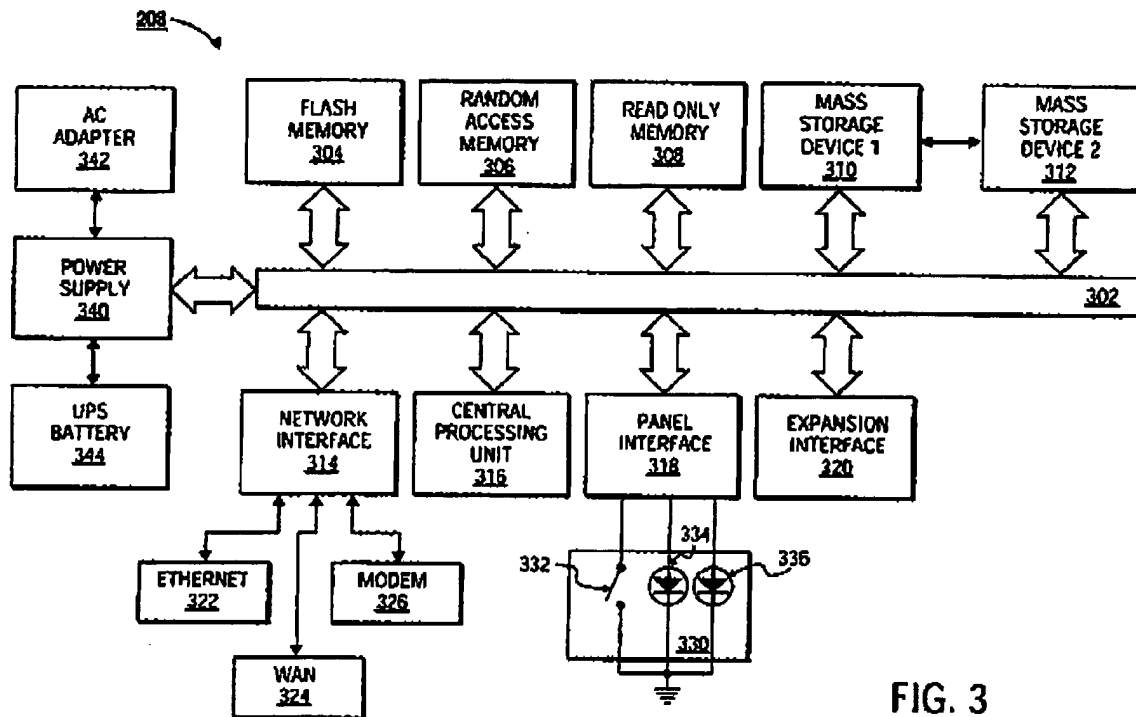


FIG. 3

The gateway interface device 208 runs several software programs in a runtime layer 406 as illustrated in Figure 5 (reproduced below). Some of these software programs include the configuration manager 506 discussed above, service managers 510, and diagnostic managers 514.

Serial No. 09/456,894

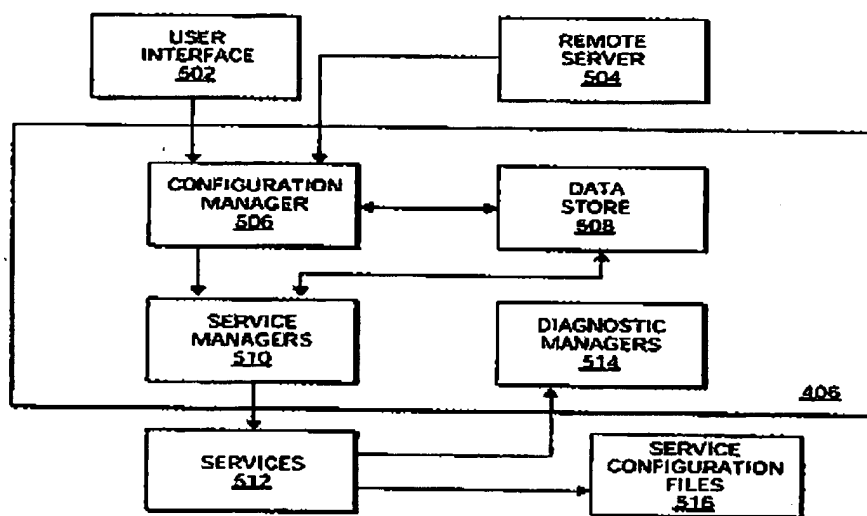


FIG. 5

The network interface device 208 is provided to connect client computers 214 to an external computer network such as the internet (as illustrated in Figure 2 reproduced above). The network interface device 208 is usually provided in an unconfigured state. The network interface device 208 is configured for the client system by automated processes and protocols initiated from the remote server 504. See Frailong reference, column 2, lines 34-45.

One of ordinary skill in the art recognizes that the Frailong reference does not teach (1) disabling access to at least a first section of code in a network driver interface, wherein the network driver interface provides for communication between one or more media access control units and one or more protocol drivers in a computer system according to a set of bindings; (2) patching the first section of code to cause the insertion of a rerouting driver into the one or more communication paths provided by the set of bindings; and (3) re-enabling access to the patched first section of code, as recited in independent Claim 1.

While the network interface 208 of Frailong reference includes a configuration manager 506 as illustrated in Figure 5 (reproduced above) which is an API operating through a remote

Serial No. 09/456,894

procedure call (RPC) protocol that communicates commands between a user interface 502 and network services 512 (see Frailong reference, column 8, lines 24-40), it is apparent to one of ordinary skill in the art that the Frailong reference is not concerned with reconfiguring code being executed by a processor. The Frailong reference simply does not teach low level operations that concern code running on a processor and patching such code to include a rerouting driver while the processor is running, as recited in independent Claim 1. The Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program.

To a lay person not skilled in the art, the Frailong reference may appear to have terminology that is similar to independent Claim 1 such as "network interface device" 208 compared to the claimed "network driver interface." However, as explained above, the network interface device 208 of Frailong is a stand alone machine and is not the same as a network driver interface that is patched with a rerouting driver as recited in independent Claim 1. Further, the Frailong reference also does not provide any teaching of media access control units, protocol drivers, and bindings, as recited in amended independent Claim 1.

Therefore, the Frailong reference simply does not teach any of the elements of independent Claim 1. In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 1. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 11

The rejection of Claim 11 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest (1) transmitting from a remote host to a first target computer on a network an installation application and a rerouting driver; (2) transmitting from the remote host to the first target computer a command to cause the first target computer to execute the installation application; (3) the first target computer, responsive to receipt of the command, executing the installation application, wherein the first target computer includes a network driver interface that provides for communication between one or more media

Serial No. 09/456,894

access control units and one or more protocol drivers according to a set of bindings; and (4) the first target computer, responsive to executing the installation application, causing the modification of the network driver interface to insert the rerouting driver into the one or more communication paths provided by the set of bindings while the network driver interface is being executed by the first target computer and without restarting the first target computer, as recited in independent Claim 11.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern code running on a processor and causing a modification of the network driver interface to insert a rerouting driver into one or more communication paths provided by a set of bindings without restarting a computer, as recited in amended independent Claim 11.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 11. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 18

The rejection of Claim 18 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: a processor for simultaneously executing: (1) a protocol driver; (2) a network driver interface; (3) a media access control unit; (4) a rerouting driver; (5) the network driver interface to store a first binding defining a communication path between the protocol driver and the media access control unit, (6) the network driver interface coupled to communicate packets with the media access control unit, (7) the network driver interface being patched to communicate the packets with the rerouting driver; and (8) the rerouting driver being executed by the processor at the same time as the network

Serial No. 09/456,894

driver interface and being coupled to communicate the packets with the protocol driver, as recited in amended independent Claim 18.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern a network driver interface coupled to communicate packets with a media access control unit and the network driver interface patched to communicate the packets with a rerouting driver. And lastly, the Frailong reference does not teach any media access control units and rerouting drivers being coupled to communicate packets with a protocol driver, as recited in amended independent Claim 18.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 18. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 23

The rejection of Claim 23 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest a rerouting driver comprising (1) control code, for controlling the rerouting driver; (2) binding code, for establishing at least one binding at the network driver interface so that the rerouting driver is bound to at least one media access control unit while the network driver interface and the rerouting driver are executed at the same time; (3) patching code, for inserting template jumps into at least a first section of code in the network driver interface, the template jumps providing jumps to templates in the rerouting driver so that information from at least one media access control unit destined for at least one protocol driver is rerouted to the rerouting driver while the network driver interface and the rerouting driver are executed at the same time; (4) at least one template, for receiving information from at

Serial No. 09/456,894

least one template jump in the network driver interface; and (5) inserted code, for evaluating rerouted information received by the template jumps, as recited in amended independent Claim 23.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern binding code, for establishing at least one binding at the network driver interface so that the rerouting driver is bound to a media access control unit; patching code, for inserting template jumps into at least a first section of code in the network driver interface, the template jumps providing jumps to templates in the rerouting driver so that information from at least one media access control unit destined for at least one protocol driver is rerouted to the rerouting driver. And lastly, the Frailong reference does not teach inserted code, for evaluating rerouted information received by the template jumps, as recited in amended independent Claim 23.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 23. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 26

The rejection of Claim 26 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest (1) selecting a first section of code of the network driver interface in a first central processing unit that is to be modified while the network driver interface is running; (2) writing the first section of code of the network driver interface into the cache memory of the first central processing unit while the network driver interface is

Serial No. 09/456,894

running; (3) overwriting a portion of the first section of code in cache memory with blocking code to create a first version of code while the network driver interface is running; (4) writing the first version of code into shared memory while the network driver interface is running; (5) modifying the first version of code in the cache memory to create a second version of code, wherein a portion of the code following the blocking code is overwritten with template jumps to effect a static patch of the network driver interface while the network driver interface is running; (6) writing the second version of code into shared memory while the network driver interface is running; (7) modifying the second version of code in the cache memory with code to create a third version of code, wherein the blocking code is overwritten to remove the blocking code while the network driver interface is running; and (8) writing the third version of code into shared memory while the network driver interface is running, as recited in amended independent Claim 26.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern selecting a first section of code in a first central processing unit that is to be modified; writing the first section of code into the cache memory of the first central processing unit; overwriting a portion of the first section of code in cache memory with blocking code to create a first version of code; writing the first version of code into shared memory; modifying the first version of code in the cache memory to create a second version of code, wherein a portion of the code following the blocking code is overwritten with template jumps to effect a static patch of the network driver interface; and writing the second version of code into shared memory,, as recited in original independent Claim 26.

Serial No. 09/456,894

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 26. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 28

The rejection of Claim 28 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest (1) disabling access to a first section of code of a network driver interface while the network driver interface is running, the first section of code providing a communication path between a media access control unit and an application, the first section of code including a generic call; and (2) overwriting the first section of code with a second section of code while the network driver interface is running whose execution causes execution flow to be rerouted to a third section of code in a rerouting driver, the second section of code being no larger than the first section of code, (3) the third section of code, when executed and while the network driver interface is running, completing the communication path and returning execution flow, the third section of code including additional code not present in the first section of code that is now inserted into the communication path, as recited in amended independent Claim 28.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern disabling and overwriting code as it is being executed, as recited in amended independent Claim 28.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or

Serial No. 09/456,894

render obvious the recitations as set forth in amended independent Claim 28. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 30

The rejection of Claim 30 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest, computers comprising: (1) a processor for running a network driver interface and the distributed packet based security system and for installing first and second code while the network driver interface is running; (2) a shared memory buffer between a user space that stores the first code of the distributed packet based security system and a system address space that stores the protocol driver and second code of the distributed packet based security system, wherein (3) said second code is coupled to said shared memory to store information regarding packets received over the network, and wherein said first code is coupled to the shared memory buffer to evaluate information stored in the shared memory buffer, as recited in amended independent Claim 30.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. The Frailong reference only mentions packets only in connection with TCP/IP over a network which is opposite to evaluating packets received over a network and selectively allowing transmission of packets. Further, the Frailong reference simply does not teach low level operations that concern patching techniques, as recited in amended independent Claim 30.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 30. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Serial No. 09/456,894

Independent Claim 34

The rejection of Claim 34 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest a plurality of networked computers each including: (1) a network driver interface; (2) a media access control unit coupled to the physical transmission medium of the network to extract packets from data provided across said medium; (3) a protocol driver coupled to the media access control unit via the network driver interface; and (4) filter code being installed such that the code is coupled to the network driver interface while the network driver interface is running and in between the media access control unit and the protocol driver and enabled without shutdown or restart to evaluate said packets and selectively allow continued transmission of different ones of said packets to the protocol driver, as recited in amended independent Claim 34.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference only mentions packets only in connection with TCP/IP over a network which is opposite to filter code installed in between a media access control unit and a protocol driver and enabled without shutdown or restart to evaluate the packets and selectively allow continued transmission of different ones of the packets to the protocol driver, as recited in amended independent Claim 34.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 34. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 38

The rejection of Claim 38 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: (1) distributing from a remote host across

Serial No. 09/456,894

a network to a plurality of computers code comprising a security filter to be installed by each of said plurality of computers, (2) each of said plurality of computers including routines to be executed to provide a communication path between a media access control unit coupled to the network and a protocol driver, said communication path for packets transmitted across said network; (3) transmitting from the remote host to each of the plurality of computers a command to cause each of the plurality of computers to execute said code; and (4) each of the plurality of computers responsive to said command performing, installing the code while running a network driver interface such that the code is in the communication path between the media access control unit and the protocol driver, (5) said installed code being enabled, without restart of said computer, to evaluate selectively allowing continued transmission of different ones of said packets received over said network along the communication path, as recited in amended independent Claim 38.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern installing a driver in the communication path between the media access control unit and the protocol driver, the installed driver being enabled, without restart of the computer, to evaluate selectively allowing continued transmission of different ones of the packets, as recited in amended independent Claim 38.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 38. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Serial No. 09/456,894

Independent Claim 42

The rejection of Claim 42 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: (1) distributing from a remote host across a network to a plurality of computers code comprising a security filter to be installed by each of said plurality of computers, (2) each of said plurality of computers including routines to be executed to provide a communication path between a media access control unit coupled to the network and a protocol driver, said communication path for packets transmitted across said network; (3) transmitting from the remote host to each of the plurality of computers a command to cause each of the plurality of computers to execute said code; and (4) each of the plurality of computers responsive to said command performing, installing the code while running a network driver interface such that the code is in the communication path between the media access control unit and the protocol driver, (5) said installed code being enabled, without restart of said computer, to evaluate selectively allowing continued transmission of different ones of said packets received over said network along the communication path, as recited in amended independent Claim 42.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern installing a driver in the communication path between the media access control unit and the protocol driver, the installed driver being enabled, without restart of the computer, to evaluate selectively allowing continued transmission of different ones of the packets received over the network, as recited in amended independent Claim 42.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 42. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Serial No. 09/456,894

Independent Claim 45

The rejection of Claim 45 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: (1) installing into each of a plurality of computers on a network code coupled to the network driver interface while the network driver interface is running, the code forming part of a distributed packet security system, said code being installed such that packets transmitted across said network to a given one of said plurality of computers is received by said code before being provided to a protocol driver; (2) at least the first of said plurality of computers without being shutdown or restarted, (3) receiving a packet from said network; and (4) said code executing on said first computer selectively forwarding said packet onto the protocol driver depending upon parameters of the distributed packet base security system, as recited in amended independent Claim 45.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern installing into each of a plurality of computers on a network code that is part of a distributed packet security system, the code being installed such that packets transmitted across the network to a given one of the plurality of computers is received by the code before being provided to a protocol driver, as recited in amended independent Claim 45.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 45. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Serial No. 09/456,894

Independent Claim 48

The rejection of Claim 48 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: (1) installing and enabling, without shutdown or restart, on each of a plurality of computers on a network code coupled to a network driver interface while the network driver interface is running, the code forming part of a distributed packet security system, (2) said code being installed such that packets transmitted across said network to a given one of said plurality of computers is received by said code before being provided to a protocol driver; (3) wherein said code, when executed responsive to one of said plurality of computers receiving a packet from said network, selectively forwards said packet onto the protocol driver depending upon parameters of the distributed packet base security system, as recited in amended independent Claim 48.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern installing and enabling, without shutdown or restart, on each of a plurality of computers on a network code that is part of a distributed packet security system, the code being installed such that packets transmitted across the network to a given one of the plurality of computers is received by the code before being provided to a protocol driver, as recited in amended independent Claim 48.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 48. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Serial No. 09/456,894

Independent Claim 51

The rejection of Claim 51 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: (1) installing into each of a plurality of computers on a network first and second code that is part of a distributed packet security system, said first code being installed in a user address space, (2) said second code being installed while the network driver interface is running and being in a communication with the network driver interface of a system address space, (3) said second code being installed such that packets transmitted across said network to a given one of said plurality of computers is received by said second code before being provided to a protocol driver in said system space; (3) at least the first of said plurality of computers without being shutdown or restarted, receiving a packet from said network; (4) said second code storing at least certain information from said packet into a shared memory buffer between the user address space and the system address space; and (5) said first code accessing information from said shared memory buffer, as recited in amended independent Claim 51.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern a first code being installed in a user address space, a second code being installed in a system address space, the second code being installed such that packets transmitted across the network to a given one of the plurality of computers is received by the second code before being provided to a protocol driver in the system space, and at least the first of the plurality of computers without being shutdown or restarted, as recited in amended independent Claim 51.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or

Serial No. 09/456,894

render obvious the recitations as set forth in amended independent Claim 51. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 55

The rejection of Claim 55 is respectfully traversed. It is respectfully submitted that the Frailong reference fails to describe, teach, or suggest: (1) installing and enabling, without shutdown or restart, on each of a plurality of computers on a network first and second code that is part of a distributed packet security system, (2) said first code being installed in a user address space, said second code being installed while the network driver interface is running and being in a communication with the network driver interface of a system address space, (3) said second code being installed such that packets transmitted across said network to a given one of said plurality of computers is received by said second code before being provided to a protocol driver in said system space; (4) wherein said second code, when executed responsive to a first of said plurality of computers receiving a packet from said network, stores at least certain information from said packet into a shared memory buffer between the user address space and the system address space; and (5) wherein said first code when executed by said first computer accesses said information from said shared memory buffer, as recited in amended independent Claim 55.

As discussed above with respect to independent Claim 1, the Frailong reference is not concerned with reconfiguring code being executed by a processor of a computer on a network. Instead, the Frailong reference is concerned with a gateway interface device 208 that services computers on a network. In other words, the Frailong reference is only concerned about communications between different programs and not the operations of code contained within any single application program. Further, the Frailong reference simply does not teach low level operations that concern (1) installing and enabling, without shutdown or restart, on each of a plurality of computers on a network first and second code that is part of a distributed packet security system, (2) the first code being installed in a user address space, (3) the second code being installed in a system address space, the second code being installed such that packets transmitted across the network to a given one of the plurality of computers is received by the

Serial No. 09/456,894

second code before being provided to a protocol driver in the system space, as recited in amended independent Claim 55.

In light of the differences between the claims and the Frailong reference mentioned above, one of ordinary skill in the art recognizes that the Frailong reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 35. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Dependent Claims 2-10, 12-17, 19-22, 24-26, 29, 33, 35-37, 39-40, 43-44, 46-47, 49-50, 52-53, and 56-57

The Applicant respectfully submits that the above-identified dependent claims are allowable because the independent claims from which they depend are patentable over the cited references. The Applicant also respectfully submits that the recitations of these dependent claims are of patentable significance.

In view of the foregoing, the Applicant respectfully requests that the Examiner withdraw the pending rejections of dependent Claims 2-10, 12-17, 19-22, 24-26, 29, 33, 35-37, 39-40, 43-44, 46-47, 49-50, 52-53, and 56-57.

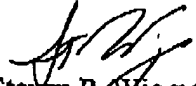
CONCLUSION

The foregoing is submitted as a full and complete response to the Office Action mailed on June 4, 2004. The Applicant and the undersigned thank Examiner El-Chanti for consideration of these remarks. The Applicant has amended the claims and has submitted remarks to traverse rejections of Claims 1-58. The Applicant respectfully submits that the present application is in condition for allowance. Such action is hereby courteously solicited.

Serial No. 09/456,894

If the Examiner believes that there are any issues that can be resolved by a telephone conference, or that there are any formalities that can be corrected by an Examiner's amendment, please contact the undersigned in the Atlanta Metropolitan area (404) 572-2884.

Respectfully submitted,


Steven P. Wigmore
Reg. No. 40,447

King & Spalding, LLP
45th Floor
191 Peachtree Street, N.E.
Atlanta, Georgia 30303
404.572.4600
K&S Docket: 05456.105030